

SYSTEM AND METHOD FOR IDENTIFYING OPEN  
PERIPHERAL COMPONENT INTERCONNECT (PCI) SLOTS

CROSS-REFERENCE TO OTHER APPLICATIONS

This application shares a common specification with U.S.  
Application No. \_\_\_\_\_ filed on January 31, 2001 for a  
5 "System and Method for Assisting in Product Selection."

TECHNICAL FIELD OF THE INVENTION

This invention relates generally to computer systems, and  
10 more particularly to a system and method for identifying open  
peripheral component interconnect (PCI) slots.

BACKGROUND OF THE INVENTION

Computing devices routinely include one or more Peripheral Component Interconnect (PCI) slots. PCI slots are interfaces used by the computer to communicate with peripheral devices, such as Ethernet cards or sound cards, which are inserted or plugged into the PCI slots. The PCI slots typically provide power to the peripheral devices and facilitate communication between the peripheral devices and other components of the computing device.

In order to obtain and install new peripheral devices in the computer, the user often needs to know whether the computing device has any "open" or available PCI slots. For example, a customer who wants to receive Digital Subscriber Line (DSL) service typically needs to obtain a DSL modem, but different DSL modems may communicate with different types of interfaces. The customer needs to determine whether the customer's computer has an available PCI slot. If a PCI slot is available, the customer can order and install a PCI-based DSL modem. Otherwise, the customer needs to order a different type of modem.

Computer users typically have difficulty in identifying whether their computing devices have available PCI slots. For example, identifying available PCI slots typically requires a physical inspection of the computer. Before identifying whether any PCI slots are available, the user often must physically remove at least a portion of the computer casing and locate the PCI slots. It is often a time-consuming process to remove the computer casing, identify the PCI slots, determine if a PCI slot is available, and replace the computer casing. Also, the computer user typically does not possess the technical knowledge needed to identify the available PCI slots. The user may be unaware of what PCI slots look like,

or the user may be unable to distinguish PCI slots from other types of interfaces.

One approach to identifying available PCI slots in a computer is to provide identification software to the user.

- 5 The software helps the user identify the presence or absence of PCI slots by, for example, illustrating the physical steps that the user must take to locate and identify the PCI slots. A problem with this approach is that the user still needs to manually examine the computer, which is a time-consuming process. Also, even with the use of identification software, the user may still incorrectly identify a PCI slot, or the user may still be unable to identify whether a PCI slot is available.

SUMMARY OF THE INVENTION

The present invention recognizes a need for an improved system and method for identifying open peripheral component interconnect (PCI) slots, which reduce or eliminate some or all of the problems and disadvantages associated with prior systems and methods.

In one embodiment of the invention, a system for identifying an available peripheral component interconnect (PCI) slot in a computing device includes a memory operable to store information identifying at least one PCI slot in the computing device. The system also includes a processor coupled to the memory and operable to identify any PCI devices coupled to a PCI bus. The PCI bus is coupled to the PCI slot. The processor is also operable to determine if any of the identified PCI slots are available without requiring physical inspection of the PCI slots. An available PCI slot comprises an identified PCI slot that is not coupled to an identified PCI device.

In another embodiment of the invention, a method for identifying an available peripheral component interconnect (PCI) slot in a computing device includes identifying at least one PCI slot in the computing device. The method also includes identifying any PCI devices coupled to a PCI bus. The PCI bus is coupled to the PCI slot. The method further includes determining if any of the identified PCI slots are available without requiring physical inspection of the PCI slots. An available PCI slot comprises an identified PCI slot that is not coupled to an identified PCI device.

Numerous technical advantages can be gained through various embodiments of the invention. Various embodiments of the invention may exhibit none, some, or all of the following advantages. For example, in one embodiment of the invention,

a system is provided that simplifies the identification of one or more open PCI slots in a computing device. Rather than requiring a physical inspection of the computing device, the system may detect and report the existence of an open PCI slot to a user. The user may not need to remove the computer's casing, identify the PCI slots, determine if a PCI slot is available, and replace the computer casing. By reducing or eliminating the need for the user to physically inspect the computing device, the system simplifies the identification of open PCI slots and reduces the time it takes to identify whether a PCI slot is available.

Some embodiments of the invention also reduce or eliminate the likelihood that the user will incorrectly identify an open PCI slot. Because the system detects and reports the existence of an open PCI slot, the user may not need to physically inspect the computing device. The user need not know what PCI slots look like or be able to distinguish PCI slots from other types of interfaces. By reducing or eliminating the need for the user to physically inspect the computing device, the system reduces the likelihood that the user will incorrectly identify an open PCI slot.

Other technical advantages will be readily apparent to one of skill in the art from the attached figures, description, and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following descriptions taken in connection with the accompanying drawings, in which:

FIGURE 1 is a block diagram illustrating an exemplary system for identifying open peripheral component interconnect (PCI) slots constructed according to the teachings of the present invention;

FIGURE 2 is a block diagram illustrating an exemplary PCI identification table constructed according to the teachings of the present invention;

FIGURE 3 is a block diagram illustrating an exemplary PCI enumeration table constructed according to the teachings of the present invention;

FIGURE 4 is a flow diagram illustrating an exemplary method for identifying open PCI slots according to the teachings of the present invention;

FIGURE 5 is a flow diagram illustrating one particular example of a method for identifying all PCI slots in a computing device according to the teachings of the present invention;

FIGURE 6 is a flow diagram illustrating one particular example of a method for determining whether identified PCI slots are open according to the teachings of the present invention;

FIGURE 7 is a flow diagram illustrating an exemplary method for identifying a product for use with a computing device according to the teachings of the present invention; and

FIGURE 8 is a block diagram illustrating an exemplary system for identifying a product for use with a computing

device constructed according to the teachings of the present invention.

11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2050  
2051  
2052  
2053  
2054  
2055  
2056  
2057  
2058  
2059  
2060  
2061  
2062  
2063  
2064  
2065  
2066  
2067  
2068  
2069  
2070  
2071  
2072  
2073  
2074  
2075  
2076  
2077  
2078  
2079  
2080  
2081  
2082  
2083  
2084  
2085  
2086  
2087  
2088  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2160  
2161  
2162  
2163  
2164  
2165  
2166  
2167  
2168  
2169  
2170  
2171  
2172  
2173  
2174  
2175  
2176  
2177  
2178  
2179  
2180  
2181  
2182  
2183  
2184  
2185  
2186  
2187  
2188  
2189  
2190  
2191  
2192  
2193  
2194  
2195  
2196  
2197  
2198  
2199  
2200  
2201  
2202  
2203  
2204  
2205  
2206  
2207  
2208  
2209  
2210  
2211  
2212  
2213  
2214  
2215  
2216  
2217  
2218  
2219  
2220  
2221  
222

DETAILED DESCRIPTION OF THE INVENTION

FIGURE 1 is a block diagram illustrating an exemplary system 100 for identifying open peripheral component interconnect (PCI) slots 130. In the illustrated embodiment, system 100 includes a host device 112 having one or more PCI slots 130. In one aspect of the invention, system 100 generates a PCI identification table 138, which identifies all PCI slots 130a-130n in host 112. System 100 also generates a PCI enumeration table 140, which identifies the PCI slots 130a-130n that are coupled to PCI devices. System 100 uses PCI enumeration table 140 to determine which PCI slots 130 listed in PCI identification table 138 are coupled to PCI devices. The PCI slots 130 that are listed in PCI identification table 138 but not in PCI enumeration table 140 are "open" slots, or slots that are available to be coupled to a PCI device in host 112.

Host 112 may execute with any of the well-known MS-DOS, PC-DOS, OS-2, MAC-OS, WINDOWS, UNIX, or other appropriate operating systems. Host 112 may comprise, for example, a desktop computer, a laptop computer, or any other computing or communicating device. In the illustrated embodiment, host 112 includes an input device 114, an output device 116, a hard drive 118, an optical storage media 120 or other appropriate volatile or nonvolatile storage and retrieval devices, a processor (CPU) 122, a read-only memory (ROM) 124, a random access memory (RAM) 126, and one or more controllers 128. Other embodiments of host 112 may be used without departing from the scope of the present invention.

Input device 114 may comprise, for example, a keyboard, mouse, graphics tablet, touch screen, pressure-sensitive pad, joystick, light pen, microphone, or other suitable input device. Output device 116 may comprise, for example, a video



display, a printer, a disk drive, a plotter, a speaker, or other suitable output device.

Processor 122 is coupled to read-only memory 124, random-access memory 126, and controllers 128 by a system bus 142.

5 In this document, the term "couple" refers to any direct or indirect communication between two or more elements, whether or not those elements are in physical contact with one another. Processor 122 is operable to receive information over system bus 142, process the information, and/or  
10 communicate information over system bus 142. Processor 122 may comprise any suitable processor or processors.

Read-only memory 124 is coupled to system bus 142. Read-only memory 124 stores information in system 100, such as data or instructions for processor 122. In one embodiment, read-only memory 124 includes Basic Input/Output System (BIOS)  
15 software and information, such as a PCI Interrupt Request (IRQ) routing table 136. Read-only memory 124 may comprise any of a variety of data structures, arrangements, and/or compilations operable to store and facilitate retrieval of  
20 information. Read-only memory 124 may, for example, comprise a nonvolatile memory.

Random-access memory 126 is coupled to system bus 142. Random-access memory 126 stores information in system 100, such as data used by processor 122. Random-access memory 126  
25 may comprise any of a variety of data structures, arrangements, and/or compilations operable to store and facilitate retrieval of information. Random-access memory 126 may, for example, comprise a volatile memory.

One or more controllers 128 are coupled to system bus  
30 142. Controllers 128 facilitate communication between system bus 142 and one or more peripheral devices coupled to PCI slots 130, a Local Area Network (LAN) interface 132, and/or

Universal Serial Bus (USB) ports 134. In this document, the phrase "peripheral device" refers to any hardware, software, and/or firmware operable to be coupled to a host 112. Peripheral devices include printers, monitors, CD drives, DVD drives, mice, keyboards, sound cards, modems, and any other suitable devices. Controllers 128 may, for example, receive information over a PCI bus 144 from a peripheral device coupled to a PCI slot 130. Controllers 128 may also receive information over system bus 142 and communicate the information over a USB bus 146 to a peripheral device coupled to a USB port 134. One controller 128 may be used to communicate over PCI bus 144, USB bus 146, and/or other busses, or separate controllers 128 may be used. Controllers 128 may comprise any hardware, software, firmware, or combination thereof operable to facilitate communication between host 112 and a peripheral device. In one embodiment, controllers 128 include a PCI controller for communicating over PCI bus 144 and a USB controller for communicating over USB bus 146. Controllers 128 could also include any other suitable controller, such as a Small Computer System Interface (SCSI) controller.

PCI slots 130 are coupled to PCI bus 144. Each PCI slot 130 is operable to be coupled to a PCI device. In this document, the phrase "PCI device" refers to any hardware, software, and/or firmware operable to communicate over PCI bus 144. PCI devices include printers, monitors, CD drives, DVD drives, mice, keyboards, sound cards, modems, hard drives, or any other suitable devices. PCI slot 130 also facilitates communication between the PCI device and PCI bus 144. PCI slot 130 may comprise any hardware, software, firmware, or combination thereof operable to facilitate communication between a PCI device and PCI bus 144.

LAN interface 132 is coupled to controllers 128. LAN interface 132 facilitates communication between controllers 128 and a network. LAN interface 132 may comprise any hardware, software, firmware, or combination thereof operable to facilitate communication between host 112 and a network. LAN interface 132 may, for example, comprise an Ethernet card.

USB ports 134 are coupled to USB bus 146. USB port 134 is operable to be coupled to one or more peripheral devices and to facilitate communication between the peripheral devices and USB bus 146. USB port 134 may comprise any hardware, software, firmware, or combination thereof operable to facilitate communication between a peripheral device and USB bus 146.

System bus 142 is coupled to processor 122, read-only memory 124, random access memory 126, and controllers 128. System bus 142 facilitates communication between processor 122, read-only memory 124, random access memory 126, and controllers 128. System bus 142 may, for example, transport data and instructions between processor 122 and controllers 128. System bus 142 may comprise any suitable bus operable to transfer information between elements in host 112.

PCI bus 144 is coupled to controllers 128 and one or more PCI slots 130. PCI bus 144 is a local bus that facilitates communication between controllers 128 and peripheral PCI devices coupled to PCI slots 130. PCI bus 144 may comprise any suitable local bus operable to transfer information between controllers 128 and PCI slots 130.

USB bus 146 is coupled to controllers 128 and one or more USB ports 134. USB bus 146 is a bus that facilitates communication between controllers 128 and external peripheral devices through USB ports 134. USB bus 146 may comprise any

suitable bus operable to transfer information between controllers 128 and USB ports 134.

In one aspect of operation, read-only memory 124 includes PCI Interrupt Request (IRQ) routing table 136. PCI IRQ routing table 136 includes information that identifies each PCI slot 130 in host 112. PCI IRQ routing table 136 may, for example, include information identifying PCI bus 144, a device number for each PCI slot 130 coupled to PCI bus 144, and a slot number for each PCI slot 130 coupled to PCI bus 144. In one embodiment, PCI IRQ routing table 136 is identified by a "\$PIR" string at the beginning of the table. In that particular example, PCI IRQ routing table 136 is stored in read-only memory 124 between memory addresses F0000h and FFFFFh (hexadecimal). Other identifying strings and/or memory locations may be used without departing from the scope of the present invention.

To determine whether any PCI slots 130 are available in host 112, processor 122 generates a PCI identification table 138 and a PCI enumeration table 140. To generate PCI identification table 138, processor 122 locates PCI IRQ routing table 136 in read-only memory 124. Processor 122 may, for example, scan read-only memory 124 for a "\$PIR" string, which represents the beginning of PCI IRQ routing table 136. Processor 122 accesses the information in PCI IRQ routing table 136, extracts the information about PCI slots 130, and generates PCI identification table 138. The information stored in PCI identification table 138 identifies PCI slots 130a-130n in host 112. To generate PCI enumeration table 140, controllers 128 enumerate PCI devices coupled to PCI bus 144, which includes identifying PCI slots 130 that are coupled to PCI devices. In one embodiment, host 112 executes with a WINDOWS NT operating system, and system 100 enumerates PCI

devices coupled to PCI bus 144 using a HalGetBusData function. In another embodiment, host 112 executes with a WINDOWS 98 operating system, and system 100 enumerates PCI devices using a system registry. Other methods of enumerating devices on  
5 PCI bus 144 may be used without departing from the scope of the present invention. After enumerating devices coupled to PCI bus 144, processor 122 compares PCI identification table 138 and PCI enumeration table 140. Processor 122 identifies which PCI slots 130 are not coupled to PCI devices, and these  
10 PCI slots 130 are "open" or available to be coupled to a PCI device.

The ability to identify open PCI slots 130 may be useful in many ways, such as in determining what product to install in computer 112. In this document, the term "product" refers  
15 to any hardware, software, and/or firmware that may be coupled to or used in conjunction with host 112. For example, a modem could be coupled to a PCI slot 130, LAN interface 132, or a USB port 134. In one embodiment, processor 122 may identify whether host 112 includes any available PCI slots 130.  
20 Processor 122 may also determine whether host 112 includes a LAN interface 132 and/or a USB port 134. Using this information, processor 122 may recommend the type of modem that may be installed in host 112. In one embodiment, processor 122 may recommend a USB modem if host 112 supports  
25 the USB standard. If not, processor 122 may determine whether any PCI slots 130 are open and recommend a PCI-based modem if a PCI slot 130 is available. Otherwise, processor 122 may determine whether a LAN interface 132 is present and recommend a LAN-based modem. If processor 122 cannot identify a  
30 suitable type of product, processor 122 may instruct the user to contact a technician or customer support personnel.

Although FIGURE 1 illustrates one embodiment of system 100, various changes may be made without departing from the scope of the present invention. For example, LAN interface 132 could be coupled to USB bus 146 instead of PCI bus 144. Also, any number of PCI slots 130, LAN interfaces 132, and/or USB ports 134 may be used in host 112. Further, PCI IRQ routing table 136, PCI identification table 138, and/or PCI enumeration table 140 may reside in any memory internal to or accessible by host 112. In addition, although system 100 illustrates host 112 as comprising a desktop computer, host 112 may comprise any suitable computing device, such as a laptop computer or server. Other changes may be made without departing from the scope of the present invention.

FIGURE 2 is a block diagram illustrating an exemplary PCI identification table 238. In this document, the term "table" refers to any arrangement, compilation, and/or data structure operable to store and facilitate retrieval of information. PCI identification table 238 may be useful, for example, in system 100 of FIGURE 1. In the illustrated embodiment, PCI identification table 238 includes at least one entry 250. An entry 250 in PCI identification table 238 corresponds to a PCI slot 130 in system 100, and each entry 250 includes a bus number 252, a device number 254, and a slot number 256. The values and number of entries 250 contained in PCI identification table 238 are for illustration only. Any suitable number of entries 250 and any suitable values may be used without departing from the scope of the present invention.

Bus number 252 identifies PCI bus 144 in system 100. In the illustrated embodiment, PCI bus 144 is identified by the number zero, although PCI bus 144 may be identified using any suitable indicator. Also, if multiple PCI busses 144 are used

in system 100, each bus 144 may be identified by a different bus number 250.

Device number 254 identifies a device number assigned to each PCI slot 130 in host 112. A device number 254 may be used, for example, to address a device coupled to PCI slot 130. In one embodiment, device numbers 254 are assigned to PCI slots 130 by the BIOS of host 112. PCI slots 130 may be assigned any suitable device number 254. In one embodiment, different device numbers 254 may be used to identify the same PCI slot 130. For example, when a single hardware device performs multiple functions, such as a sound card having a SCSI controller, two device numbers 254 could be assigned to the PCI slot 130 that is coupled to the sound card.

Slot number 256 identifies whether an entry 250 in PCI identification table 238 identifies a system-board device or a PCI slot 130. A slot number 256 of zero indicates that entry 250 corresponds to a system-board device, or a device that is hardwired to the motherboard of host 112. A slot number 256 having a non-zero value represents an "add-in slot," or a PCI slot 130 in host 112 that can be coupled to a PCI device. In one embodiment, an entry 250 having a slot number 256 of zero is not an available PCI slot 130, and an entry 250 having a non-zero slot number 256 may or may not be an available PCI slot 130.

In one embodiment, processor 122 generates PCI identification table 238 using information from PCI IRQ routing table 136. For example, processor 122 may scan read-only memory 124 for a "\$PIR" string, which represents the beginning of PCI IRQ routing table 136. After locating table 136 in read-only memory 124, processor 122 may extract the bus number 252, device number 254, and slot number 256 for each

entry in PCI IRQ routing table 136. Using this information, processor 122 may generate PCI identification table 238.

Although FIGURE 2 illustrates one embodiment of PCI identification table 238, other embodiments may be used without departing from the scope of the present invention. For example, although FIGURE 2 illustrates a table 238 storing the information in entries 250, other data structures may be used in system 100. The information from entries 250 could, for example, be stored in a linked list, multiple arrays, or any other suitable data structure operable to store and facilitate retrieval of information.

FIGURE 3 is a block diagram illustrating an exemplary PCI enumeration table 340. PCI enumeration table 340 may be useful, for example, in system 100 of FIGURE 1. In the illustrated embodiment, PCI enumeration table 340 includes zero or more entries 350. An entry 350 in PCI enumeration table 340 corresponds to a PCI device coupled to PCI bus 144 in system 100. Each entry 350 in PCI enumeration table 340 includes a bus number 352 and a device number 354. The values and number of entries 350 contained in PCI enumeration table 340 are for illustration only. Any suitable number of entries 350 and any suitable values may be used without departing from the scope of the present invention.

Bus number 352 identifies PCI bus 144 in system 100. Device number 354 identifies a device number assigned to a PCI device coupled to PCI bus 144. Device numbers 354 in entries 350 may have any suitable value, and different device numbers 354 may be used to identify the same PCI device.

In one embodiment, processor 122 generates PCI enumeration table 340 using information collected by controllers 128. For example, controllers 128 may enumerate PCI devices coupled to PCI bus 144 using a HalGetBusData



function or a system registry. During enumeration, controllers 128 identify the PCI devices coupled to PCI bus 144. For each device, controllers 128 identify the bus number 352 of the device and the device number 354 of the device.

- 5 Processor 122 collects this information and generates PCI enumeration table 340.

Although FIGURE 3 illustrates a table 340 storing the information in entries 350, other data structures may be used to store the information without departing from the scope of the present invention. The information from entries 350 could, for example, be stored in a linked list, multiple arrays, or any other suitable data structure operable to store and facilitate retrieval of information. Also, because the number of entries 350 in PCI enumeration table 340 is related to the number of PCI devices coupled to PCI bus 144, table 340 may contain any number of entries 350.

In one aspect of operation, processor 122 generates PCI identification table 238 and PCI enumeration table 340. To identify open PCI slots 130 in host 112, processor 122 may compare entries 250 and 350 in tables 238 and 340. In one embodiment, processor 122 compares the bus number 352 and device number 354 of an entry 350 in PCI enumeration table 340 with the entries 250 in PCI identification table. If an entry 250 in table 238 has a bus number 252 and a device number 254 that matches the bus number 352 and device number 354 of entry 350 in table 340, entry 250 is removed from table 238. In this instance, entry 250 does not correspond to an open PCI slot 130 because an enumerated PCI device is present in the PCI slot 130. Processor 122 repeats this process for each entry 350 in table 340, and any remaining entries 250 in table 238 represent open PCI slots 130 in system 100. Alternatively, instead of removing entries 250 from table 238,

system 100 could construct a new table that stores information identifying open PCI slots 130.

Using the tables of FIGURES 2 and 3 as an example, entry 350k has a bus number 352 of "0" and a device number 354 of "f". Processor 122 compares bus number 352 and device number 354 to entries 250 in table 238 of FIGURE 2, and processor 122 finds that entry 250l in table 238 has a bus number 252 of "0" and a device number 254 of "f". Entry 250l is removed from table 238 because it is not an available PCI slot 130. After repeating this process for each entry 350, only entry 250k would remain in table 238. Entry 250k corresponds to slot 3, which is an available PCI slot 130. Because slot 3 is present in host 112 but is not coupled to a PCI device, slot 3 is an open PCI slot 130.

In the illustrated embodiment, entry 250n has a slot number 256 of zero. An entry 250 having a slot number 256 of zero corresponds to a hardwired device on the motherboard of host 112. That entry 250 does not represent an available PCI slot 130. In one embodiment, processor 122 could allow entry 250n to remain in table 238. Entry 250n would eventually be removed from table 238 when processor 122 compares it to entry 350m. In another embodiment, system 100 could remove entries 250 having a slot number 256 of zero before comparing entries 250 and 350.

In the illustrated embodiment, entry 350a in table 340 contains values of zero for bus number 352 and device number 354. In one embodiment, devices such as a PCI bus controller 128 may be enumerated as a device on PCI bus 144. In a particular embodiment, controller 128 may be enumerated as having a bus number 352 of zero and a device number 354 of zero. The controller 128 is not a physical PCI slot 130 in

host 112, so system 100 may ignore this entry 350 in table 340 when identifying open PCI slots 130 in host 112.

FIGURE 4 is a block diagram illustrating an exemplary method 400 for identifying open PCI slots 130. System 100 identifies PCI slots 130 in host 112 at step 500. This may include, for example, processor 122 scanning read-only memory 124 and locating PCI IRQ routing table 136. This may also include processor 122 extracting a bus number 252 and a device number 254 for each PCI slot 130 from PCI IRQ routing table 136, and storing the bus numbers 252 and device numbers 254 in PCI identification table 238. System 100 identifies PCI devices coupled to PCI bus 144 at step 550. This may include, for example, controllers 128 enumerating the PCI devices by identifying a bus number 352 and a device number 354 of any PCI device coupled to PCI bus 144. This may also include processor 122 storing the bus numbers 352 and device numbers 354 in PCI enumeration table 340.

System 100 determines, without requiring physical inspection of the PCI slots 130, which identified PCI slots 130 are available at step 600. This may include, for example, processor 122 comparing the entries 250 in PCI identification table 238 and the entries 350 in PCI enumeration table 340. This may also include processor 122 identifying any entry 250 in PCI identification table 238 that does not have a matching entry 350 in PCI enumeration table 340.

Although FIGURE 4 illustrates one embodiment of a method 400 for identifying open PCI slots 130, various changes may be made to method 400 without departing from the scope of the present invention. For example, system 100 may identify PCI devices coupled to PCI bus 144 before identifying PCI slots 130 in host 112.

FIGURE 5 is a flow diagram illustrating one particular example of a method 500 for identifying all PCI slots 130 in a computing device 112. System 100 scans a memory for a PCI IRQ routing table 136 at step 502. This may include, for example, processor 122 scanning read-only memory 124 between memory addresses F0000h and FFFFFh (hexadecimal). System 100 locates the PCI IRQ routing table 136 at step 504. This may include, for example, processor 122 locating a "\$PIR" string in read-only memory 124.

System 100 determines the number of entries in PCI IRQ routing table 136 at step 506. In one embodiment, PCI IRQ routing table 136 includes an entry that defines the size of PCI IRQ routing table 136 in bytes. Also, PCI IRQ routing table 136 may contain thirty-two bytes of information in addition to the entries, and each entry may contain sixteen bytes of information. In that particular example, processor 122 may determine the number of entries in PCI IRQ routing table 136 using the formula:

$$\text{Number of entries} = (\text{Size of table} - 32) / 16.$$

The number of entries represents the number of PCI slots 130 and system-board devices coupled to PCI bus 144 in host 112.

System 100 identifies the bus number 252 and device number 254 for each entry at step 508. This may include, for example, processor 122 scanning PCI IRQ routing table 136 and extracting the bus number 252 and device number 254 for each entry. System 100 may store this information in PCI identification table 238 or any other suitable data structure.

Although FIGURE 5 illustrates one embodiment of a method 500 for identifying PCI slots 130 in a host 112, various changes may be made to method 500 without departing from the

scope of the present invention. For example, system 100 may identify the number of entries in PCI IRQ routing table 136 during or after the scan of table 136, such as by counting the number of entries found in table 236. Also, system 100 may  
5 extract additional information about each PCI slot 130 from PCI IRQ routing table 136, such as a slot number 256 for each entry.

FIGURE 6 is a flow diagram illustrating one particular example of a method 600 for determining whether identified PCI slots 130 are open. System 100 initializes an open slot counter at step 602. This may include, for example, processor 122 setting the counter to a value equal to the number of entries 250 in table 238. System 100 removes any identified PCI slots 130 having a slot number of zero at step 604. This  
10 may include, for example, processor 122 scanning table 238 and removing any entries 250 having a slot number 256 of zero since these entries 250 do not correspond to an open PCI slot 130. This may also include processor 122 decrementing the open slot counter for each entry 250 removed. System 100  
15 selects the first enumerated PCI device at step 606. This may include, for example, processor 122 selecting the first entry 350 in PCI enumeration table 340.

System 100 compares the enumerated PCI device to the identified PCI slots 130 at step 608. This may include, for  
25 example, processor 122 comparing the bus number 352 and device number 354 of entry 350 to the bus numbers 252 and device numbers 254 of entries 250. If system 100 finds an entry 250 having a bus number 252 and device number 254 that matches the bus number 352 and device number 354 of entry 350, system 100  
30 identifies a match at step 609. In that case, the PCI slot 130 corresponding to entry 250 is not an available slot because the PCI slot 130 is coupled to a PCI device. System

100 removes the identified PCI slot 130 at step 610. This may include, for example, processor 122 removing entry 250 from table 238. System 100 also decrements the open slot counter at step 612.

5 After decrementing the open slot counter at step 612, or if system 100 found no match at step 609, system 100 determines whether another enumerated PCI device exists at step 614. This may include, for example, processor 122 determining whether another entry 350 exists in PCI enumeration table 340. If another PCI device exists, system 100 selects the next PCI device at step 616 and returns to step 608 to process this PCI device.

10 If no more enumerated PCI devices exist at step 614, method 600 ends. At this point, any remaining entries 250 in table 238 identify open or available PCI slots 130. Also, the value of the open slot counter should equal the number of available PCI slots 130.

15 Although FIGURE 6 illustrates one embodiment of a method 600 for determining whether identified PCI slots 130 are open, various changes may be made without departing from the scope of the present invention. For example, entries 250 and 350 in tables 238 and 340 could be sorted in order of increasing or decreasing device number 254 and 354 to simplify the determination at step 609. Also, in another embodiment, step 20 604 could be omitted because devices having a slot number 256 of zero should appear in PCI enumeration table 340 as enumerated PCI devices. Further, in the illustrated embodiment, system 100 compares the bus number 352 and device number 354 of each enumerated PCI device to the bus numbers 25 252 and device numbers 254 of identified PCI slots 130. In another embodiment, system 100 may compare the bus number 252 and device number 254 of each identified PCI slot 130 to the

bus numbers 352 and device numbers 354 of enumerated PCI devices.

FIGURE 7 is a flow diagram illustrating an exemplary method 700 for identifying a product for use with a computing device 112. The method 700 begins at step 702 where system 100 identifies the operating system of host 112. This may include, for example, processor 122 determining whether host 112 is executing with WINDOWS 98 or WINDOWS 98 Second Edition, or whether host 112 is executing with another operating system. System 100 determines whether the operating system of host 112 supports the use of a USB bus 146 at step 704. For example, the WINDOWS 98 operating system supports a USB bus 146, while the WINDOWS 95, WINDOWS NT 3.51, and WINDOWS NT 4.0 operating systems do not.

If the operating system supports the use of a USB bus 146, system 100 determines if a USB root hub is present at step 706. This may include, for example, processor 122 accessing a system registry to determine if a USB root hub is installed on the motherboard of host 112 or attached to host 112 as an adapter card. If a USB root hub is present, system 100 identifies a USB-based product as available for recommendation at step 708. Alternatively, system 100 could automatically recommend a USB-based product at this point. A USB device may be coupled to a USB port 134 and used with host 112.

System 100 can also determine whether any PCI slots 130 are available at step 710. This may include any method of determining whether a PCI slot 130 is open, such as processor 122 executing method 400 illustrated in FIGURE 4. If any PCI slots 130 are available, system 100 identifies a PCI-based product as available for recommendation at step 712. Alternatively, system 100 could automatically recommend a PCI-

based product at this point. A PCI device may be coupled to a PCI slot 130 and used with host 112.

System 100 can further determine if a LAN interface 132 is available for use in host 112 at step 714. This may include, for example, system 100 prompting a user to enter whether host 112 includes a LAN interface 132, or processor 122 automatically detecting the presence or absence of LAN interface 132. If a LAN interface 132 is present, system 100 identifies a LAN-based product as available for recommendation at step 716. Alternatively, system 100 could automatically recommend a LAN-based product at this point. If no product has been identified, method 700 ends without recommending a product. A user of host 112 may be advised to contact a technician or customer support personnel for assistance in identifying the type of product to install in host 112.

Although FIGURE 7 illustrates one embodiment of a method 700 for identifying a product for use with a computing device 112, various changes may be made to method 700 without departing from the scope of the present invention. For example, system 100 may perform testing steps 704, 706, 710, and 714 in any order. In another embodiment, system 100 may determine whether a LAN interface 132 is available before testing whether any PCI slots 130 are available. Also, while FIGURE 7 illustrates system 100 testing for three types of interfaces, system 100 may check whether any number of interfaces are available. For example, system 100 may determine whether any USB ports 134 or PCI slots 130 are available, without determining if a LAN interface 132 is present. In addition, system 100 may take any suitable action after identifying one or more available interfaces. In one embodiment, system 100 may present a user with a list of all identified products and allow the user to select one or more



of the products. In another embodiment, system 100 may recommend one or more of the identified products to the user using any suitable ranking or rating criteria. Other changes may be made to method 700 without departing from the scope of the present invention.

The process described in FIGURE 7 could be hardwired into host 112 or could exist all or in part as an application 148 residing on host 112 or otherwise accessible to host 112. Where all or a part of the process is executed through application 148, host 112 could access application 148, for example, by loading application 148 from a CD, a floppy disk, or any other computer readable medium. Alternatively, host 112 could access an application 148 residing at a remote site, such as a network server.

FIGURE 8 is a block diagram illustrating an exemplary system 800 for identifying a product for use with a computing device 812. In the illustrated embodiment, system 800 includes a host 812 and a server 850 coupled to a network 852. Other embodiments of system 800 may be used without departing from the scope of the present invention.

Host 812 and server 850 are coupled to network 852. Host 812 may be the same or similar to host 112 of FIGURE 1. Server 850 is operable to communicate with host 812 and to identify a product for use with host 812. In one embodiment, an application 848 resides at server 850, and host 812 downloads and executes the application 848 to identify available interfaces in host 812 and/or identify a product for use with host 812. In one embodiment, server 850 includes a web portal 854, and a user of host 812 may access the web portal 854 through network 852. Server 850 may comprise any suitable hardware, software, firmware, or combination thereof operable to communicate with host 812 over network 852.

Network 852 is coupled to host 812 and server 850 by communication links 856. Network 852 may include any suitable wireline or wireless system that supports communication between network elements using ground-based and/or space-based components. For example, network 852 may be a public switched telephone network (PSTN), an integrated services digital network (ISDN), a local area networks (LAN), a wide area network (WAN), a global computer network such as the Internet, or any other communications system or systems at one or more locations.

Communication links 856 facilitate communication between host 812 or server 850 and network 852. Communication link 856 may comprise any communications medium operable to facilitate communication of analog and/or digital signals using ground-based and/or space-based components. Communication link 856 may, for example, comprise a twisted-pair copper telephone line or a fiber optic line.

In one embodiment, a user of host 812 may access web portal 854 of a product supplier and download an application 848 to host 812. The application 848 may be the same or similar to application 148 of FIGURE 1. The user may choose to execute the application 848, or the application 848 may be a self-executing application. The application 848 may then identify the types of interfaces available in host 812. In a particular embodiment, application 848 identifies whether any PCI slots, LAN interfaces, and/or USB ports are available in host 812. Application 848 uses this information to identify or recommend one or more products for use with host 812. The application 848 may, for example, use the same or similar method illustrated in FIGURE 7 for identifying a product.

Application 848 may take any appropriate steps after identifying the types of interfaces available in host 812.

The application 848 may, for example, present the user with a list of all identified products and allow the user to select one or more of the products. Application 848 may also recommend one of the identified products to the user, or application 848 may select a product without user input. In addition, after a product is selected, application 848 may communicate an identification of the selected product to server 850. Server 850 may then take any suitable action, such as generating an order for the product, collecting billing information from the user, estimating a delivery time for the product, and/or informing the user of the estimated delivery time.

As a particular example, host 812 may include a 56K modem 858, and the user of host 812 may wish to obtain a Digital Subscriber Line (DSL) modem from a DSL service provider. The user may access web portal 854 of the DSL service provider and download application 848 to host 812. Host 812 may execute the application 848, and the application 848 identifies the types of interfaces available in host 812. The application 848 may also identify one or more types of DSL modems for use with host 812, recommend a type of DSL modem, or automatically select a modem. Application 848 may also communicate the selected modem type to the web portal 854, and delivery of the DSL modem can be arranged. In one embodiment, application 848 may treat the interface coupled to modem 858 as an available slot and instruct the user of host 812 to insert the DSL modem into the slot occupied by the 56K modem.

Although FIGURE 8 illustrates one embodiment of a system 800 for identifying a product for use with a computing device 812, various changes may be made to system 800 without departing from the scope of the present invention. For example, although application 848 is described as identifying

and/or recommending a product for use with host 812, this functionality may be divided between application 848 and server 850. In one embodiment, application 848 may identify the types of interfaces available in host 812 and communicate this information to server 850. In this embodiment, server 850 may identify one or more products for use with host 812 and communicate this information to host 812. Also, application 848 may be delivered to host 812 using any suitable method. As an example, application 848 may be placed on a CD, which is then placed into a CD drive in host 812. Other changes may be made to system 800 without departing from the scope of the present invention.

Although the present invention has been described in several embodiments, a myriad of changes, variations, alterations, transformations, and modifications may be suggested to one skilled in the art, and it is intended that the present invention encompass such changes, variations, alterations, transformations, and modifications as fall within the spirit and scope of the appended claims.